REMARKS

Claims 1-14 and 16-53 remain pending in the application. Reconsideration is respectfully requested in light of the following remarks.

Section 103(a) Rejections:

The Examiner rejected claims 1-8, 11-14, 16-21, 24-35, 38-48 and 51-53 under 35 U.S.C. § 103(a) as being unpatentable over Lucassen et al. (U.S. Publication 2003/0023953) (hereinafter "Lucassen") in view of Boloker et al. (U.S. Publication 2002/0194388) (hereinafter "Boloker"), and claims 9, 10, 22, 23, 36, 37, 49 and 50 as being unpatentable over Lucassen and Boloker further in view of Umezu et al. (U.S. Publication 2002/0109734) (hereinafter "Umezu"). Applicants respectfully traverse these rejections for at least the reasons below.

Regarding claim 1, contrary to the Examiner's assertion, Lucassen does not teach or suggest a dynamic component generator configured to *receive a new set of requirements for the application; determine whether the new set of requirements includes changes from the initial set of requirements; and if the new set of requirements includes changes from the initial set of requirements, generate a second dynamic component to replace the first dynamic component.* The Examiner cites Lucassen, paragraphs 108, 109 and 123, where Lucassen describes dynamically generating interaction logic and presentation layers and customization at runtime. However, nowhere does Lucassen describe a dynamic component generator *determining whether a new set of requirements includes changes from an initial set of requirements.*

Lucassen teaches an application development system for multi-channel applications. Lucassen describes applications that allow a user to interface in parallel with same information via multiple channels and user interfaces, such as voice and graphics. Lucassen describes an interaction-based application framework utilizing different programming layers, such as a business logic layer, interaction logic layer, and

customization layer, to specify the multi-channel applications. Lucassen's system includes an interaction manager for generating a presentation layer (Lucassen, Abstract, paragraphs 41, 59, 67, and 105 – 109). Specifically, Lucassen states, "the interaction manager 57 receives the interaction logic layer 53 and the customization meta-data 54 and generates functional or customized presentations for a particular delivery context." Lucassen further teaches that the interaction logic layer is "an abstract description of an application that describes how a user can interact with the application" (paragraph 107) and that the customization layer includes metadata associated with the interaction logic layer to optimize the presentation that will be generated by an adaptation process for a particular delivery context (e.g. voice or graphic). Lucassen further teaches that developers use a MVC-based editor/IDE development tool including a model editor for programming the interaction logic and customization layers (paragraphs 131,134 and 137). Thus, Lucassen's application generates a presentation from a developer-generated interaction logic layer and (developer-generated) customization meta-data.

Lucassen teaches that to change the presentation views, a developer would use the development tool to "access, edit and visualize the interaction logic and customization meta-data representation" (paragraph 137). After the developer modifies the underlying interaction logic and customization meta-data, Lucassen's application would generate new, different presentations.

Lucassen's system does not include a dynamic component generator configured to determine whether a new set of requirements includes changes from an initial set of requirements. Instead, Lucassen teaches that new developer-generated interaction logic layers are used to generate new presentation layers. Thus, when a developer creates a new interaction logic layer, Lucassen's system will generate a new presentation layer accordingly. However, Lucassen's system does not include any dynamic component generator <u>determining</u> whether a new set of requirements includes <u>changes from an initial set of requirements</u>. Generating a new presentation layer based on a new interaction logic layer does not disclose or anticipate a dynamic component generator configured to

determine whether a new set of requirements includes changes from an initial set of requirements, as recited in Applicants' claim.

In the Office Action mailed August 25, 2006 the Examiner admits that **Lucassen does not specifically disclose determining whether the new set of requirements includes changes from the initial set of requirements and generating a second dynamic component to replace the first dynamic component if the new set of requirements includes changes from the initial set of requirements, and relies on Boloker to teach these limitations.** The Examiner submits that Boloker teaches a model-view-controller framework (paragraph [0061]), determining whether the new set of requirements includes changes from the initial set of requirements (Automatic adaptation of the applications based... user preferences; page 5, paragraph [0082]). Boloker is directed to systems and methods for building modular Document Object Model (DOM) based multi-modal browsers, i.e., building browsers that support multiple user interface modes. The browsers are based on a framework that includes a single information source (model), mapped to multiple views, and manipulated using multiple controllers. Each controller processes and transforms the model to create a channel-specific view (e.g., a graphical view or a speech/voice view).

The cited passages of Boloker describe the multi-modal browsers and state that they support seamless transitions in the user interaction amongst the different modalities available to the user, whether such interaction is on one or across multiple devices. The sentence included in the Examiner's remarks above from paragraph [0082] does not describe that an application is dynamically adapted based on a new set of requirements, as the Examiner suggests. Instead, this sentence, taken along with others in the same paragraph, describes that when appropriate multi-modal middleware becomes available, users may influence what information is provided and in what form it is provided based on user preferences. For example, when middleware becomes available to support it, the preferred modality may change based on the user's activity and environment, such as switching from a speech-driven banking transaction to a GUI banking transaction if another person enters a room. In the system of Boloker, various modular components of

the system (e.g., different browser modules) may be executed depending on user preferences, activities, and environmental factors. However, there is nothing in these passages or elsewhere in Boloker that describes these preferences, activities, or environmental factors as a "new set of requirements" received for an application, and for which a second dynamic component (e.g., a new module) must be generated. Instead, these factors may be considered when switching between modalities in Boloker, and this switching is done by the system dynamically choosing which modality or combination of modalities (i.e., modalities already supported in the system) best fits the user's current needs. (See, e.g., paragraph [0081].)

The Examiner further submits that Boloker teaches if the new set of requirements includes changes from the initial set of requirements, generate a second dynamic component to replace the first dynamic component (if the user agent respects the requirement set in to update its focus when instructed, the multi-modal shell 41 can update at the same time the XHTML-MP page; page 4, paragraph [0188]). This passage of Boloker, taken together with preceding and subsequent paragraphs, describes an application that is designed to collect from the user his first name, last name and address. In this example, if the user does not enter his last name, the application may update the presented input form to the user to focus the user on the "last name" input field and/or may prompt the user with a speech browser asking him "what is your last name". The Examiner's cited excerpt has nothing to do with determining if a new set of requirements for an application includes changes from an initial set of requirements. In addition, the Examiner's cited passages say nothing about generating a second dynamic component to replace the first dynamic component. In paragraph [0188], the application merely executes as it is coded to focus in the blank input field. At most, Boloker teaches dynamically executing modular components of an application, but it clearly does not teach or suggest generating a second dynamic component in response to receiving a new set of requirements that includes changes from an initial set of requirements, as recited in claim 1. Neither Lucassen nor Boloker teaches this feature of claim 1. Thus, even if the references are considered in combination they do not teach or suggest Applicants' claimed invention.

The Examiner submits that it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Boloker and Lucassen because Boloker's teaching provides the ability to dynamically update its choice of modalities based on what the user chooses to do (page 5, paragraph [0081] of Boloker). Applicants assert that even if the combination of Boloker and Lucassen taught what the Examiner's remarks describe (i.e., "dynamically **update its choice** of modalities") this does not teach the limitations of Applicants' claimed invention. Updating a choice from among a plurality of available modalities is clearly not the same as generating a second dynamic component in response to receiving a new set of requirements that includes changes from an initial set of requirements, as recited in claim 1.

Applicants remind the Examiner that in order to reject a claim as obvious, the Examiner has the burden of establishing a *prima facie* case of obviousness. *In re Warner* et al., 379 F.2d 1011, 154 U.S.P.Q. 173, 177-178 (C.C.P.A. 1967). To establish a *prima facie* obviousness of a claimed invention, all the claim limitations must be taught or suggested by the prior art. *In re Royka*, 490 F.2d 981, 180 U.S.P.Q. 580 (C.C.P.A. 1974), MPEP § 2143.03. As discussed above, Lucassen in view of Boloker clearly fails to teach or suggest all of the limitations of Applicants' claim 1.

For at least the reasons above, the rejection of claim 1 is not supported by the cited art and removal thereof is respectfully requested. Similar remarks also apply to independent claims 14, 27 and 41.

Applicants assert that numerous ones of the dependent claims recite further distinctions over the cited art. Applicants traverse the rejection of these claims for at least the reasons given above in regard to the claims from which they depend. However, since the rejections have been shown to be unsupported for the independent claims, a further discussion of the dependent claims is not necessary at this time.
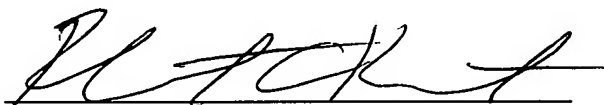
CONCLUSION

Applicants submit the application is in condition for allowance, and prompt notice to that effect is respectfully requested.

If any fees are due, the Commissioner is authorized to charge said fees to Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C. Deposit Account No. 501505/5681-08800/RCK.

Also enclosed herewith are the following items:

☒ Return Receipt Postcard

☐ Petition for Extension of Time

☐ Notice of Change of Address

☐ Other:

Respectfully submitted,

Robert C. Kowert
Reg. No. 39,255
ATTORNEY FOR APPLICANT(S)

Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C.
P.O. Box 398
Austin, TX 78767-0398
Phone: (512) 853-8850

Date: October 25, 2006